Should Models be Configuable?

Hadronic models are generally detailed implementations of specific theoretical or phenomenological descriptions of specific interactions

Contain large numbers of "adjustable" parameters

Values generally determined by original authors, or tuned to provide best match to broadest range of data

Developers may need ability to tune parameters to better match data, or to restore match if other simulation areas are changed

Allowing end users to "arbitrarily" change models essentially negates ability of GEANT4 to validate performance

Envvar vs. UIcommand

Three basic ways to change numerical parameters in models

Editing

Modify source code in local GEANT4 installation directly, rebuild toolkit, and relink (if necessary) end-user applications

Environment Variables

Use class data members (`const` or `static`) to hold parameter values, initialize in `.cc` file with **getenv()**

Macro Commands

Use a **G4UImessenger** to define commands to set values for class data members, initialize with defaults in ctor

Bertini: Singleton Container

Up to 9.6-beta, Bertini had collected a dozen adjustable parameters, overridden with envvars, spread among four source files

- Global configuration flags
- Nuclear structure parameters
- Final-state clustering parameters

Now (**hadr-casc-V09-05-15**) in singleton **G4CascadeParameters**

- Parameters have defaults, overridden by envvars in ctor
- Static accessors used by client code
- No interface to explicitly set parameter values
- Could add Messenger class with friendship